AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method for checking a model, which defines states of a hardware system under study and a transition relation among the states, the method comprising:

providing a specification of a path to be found through the states of the hardware system under study from an initial set that comprises at least one initial state among the states of the hardware system to a target set that comprises at least one target state among the states of the hardware system, such that a specified sequence of events is to occur on the path between the at least one initial state and the at least one target state;

beginning from the initial set, computing successive reachable sets comprising the states of the hardware system that are reachable from the initial set along the path, such that in the successive reachable sets the events occur in the specified sequence;

determining whether an intersection exists between one of the reachable sets on the path and the target set; and

when the intersection is not found to exist, producing a partial trace along the path between the at least one initial

state and a termination state in which at least one of the specified events occurs.

2. (Previously presented) A method according to claim 1, wherein providing the specification of the path comprises defining the events in terms of transitions among the states of the hardware system under study.

3. (Previously presented) A method according to claim 2, wherein defining the events comprises defining the transitions such that in the specified sequence of events, at least two consecutive transitions are separated by more than one cycle of the transition relation.

4. (Original) A method according to claim 2, wherein computing the successive reachable sets comprises building a non-deterministic automaton based on the transitions, and computing the reachable sets using the automaton.

5. (Currently amended) A method according to claim 4, wherein building the non-deterministic automaton comprises defining Boolean conditions corresponding respectively to the transitions, and wherein ~~detecting the~~ occurrence of the events ~~comprises~~ is detected by testing the Boolean conditions.

6. (Previously presented) A method according to claim 1, wherein computing the successive reachable sets comprises detecting occurrence of the events in the specified sequence.

7. (Original) A method according to claim 6, and comprising informing a user upon detecting occurrence of the events.

8. (Previously presented) A method according to claim 6, wherein producing the partial trace comprises choosing the termination state to be one of the states in which a final event occurs in the specified sequence of the events whose occurrence has been detected.

9. (Currently amended) A method according to claim 1, wherein computing the successive reachable sets comprises:

determining a first set among the reachable sets, disjoint from the initial set, such that all of the states in the first set are reached from the ~~initial states~~ at least one initial state in a first cycle of the transition relation; and

determining the successive reachable sets, following the first set, such that all the states in each of the sets are reached from the states in ~~the~~ a preceding set in a successive cycle of the transition relation, and so that each of the sets is disjoint from the initial set and from ~~the~~ other sets determined before it.

10. (Original) A method according to claim 9, wherein producing the partial trace comprises selecting one of the states from each of at least some of the successive reachable sets.

11. (Currently amended) A method according to claim 10, wherein selecting the one of the states comprises, for each of the ~~selected~~ states that is selected, choosing a predecessor state among the states in the preceding set until the state on the trace in the first set is found, and choosing, in the initial set, the predecessor state to the state in the first set.

12. (Original) A method according to claim 1, and comprising, when it is determined that the intersection exists between the target set and one of the reachable sets, producing a complete trace from the at least one target state through the states in the reachable sets to the at least one initial state.

13. (Original) A method according to claim 12, wherein producing the complete trace comprises computing the trace so that all the events occur along the trace in the specified sequence.

14. (Currently amended) A method according to claim 1, wherein ~~specifying~~ providing the specification of the path

comprises specifying a property to be fulfilled by the at least one target state.

15. (Previously presented) A method according to claim 14, wherein specifying the property comprises specifying a condition that is expected to be true over all of the reachable states of the hardware system under study, and wherein the condition is false in the at least one target state.

16. (Previously presented) A method according to claim 14, wherein specifying the property comprises specifying a condition representing a desired behavior of the hardware system under study, such that the condition is fulfilled in the at least one target state.

17. (Original) A method according to claim 14, wherein computing the successive reachable sets comprises testing the property while computing the sets, and ceasing to compute the sets when the intersection is found to exist.

18. (Previously presented) Model checking apparatus, comprising a model processor, which is arranged to receive a model that defines states of a hardware system under study and a transition relation among the states, and to receive a specification of a path to be found through the states of the hardware system under study from an initial set that comprises

at least one initial state among the states of the hardware system to a target set that comprises at least one target state among the states of the hardware system, such that a specified sequence of events is to occur on the path between the at least one initial state and the at least one target state, the processor being further arranged to compute, beginning from the initial set, successive reachable sets comprising the states of the hardware system that are reachable from the initial set along the path, such that in the successive reachable sets the events occur in the specified sequence, and to determine whether an intersection exists between one of the reachable sets on the path and the target set, and when the intersection is not found to exist, to produce a partial trace along the path between the at least one initial state and a termination state in which at least one of the specified events occurs.

19. (Previously presented) Apparatus according to claim 18, wherein the specification of the path comprises a definition of the events in terms of transitions among the states of the hardware system under study.

20. (Previously presented) Apparatus according to claim 19, wherein the events are defined in terms of the transitions such that in the specified sequence of events, at least two

consecutive transitions are separated by more than one cycle of the transition relation.

21. (Original) Apparatus according to claim 19, wherein the processor is arranged to build a non-deterministic automaton based on the transitions, and to compute the reachable sets using the automaton.

22. (Currently amended) Apparatus according to claim 21, wherein the processor is arranged to determine Boolean conditions corresponding respectively to the transitions, and to detect ~~the~~ occurrence of the events ~~comprises~~ by testing the Boolean conditions.

23. (Previously presented) Apparatus according to claim 18, wherein the processor is arranged to detect occurrence of the events in the specified sequence while computing the successive reachable sets.

24. (Original) Apparatus according to claim 23, wherein the processor is arranged to inform a user upon detecting occurrence of the events.

25. (Previously presented) Apparatus according to claim 23, wherein to produce the partial trace, the processor is arranged to choose the termination state to be one of the states in which a final event occurs in the specified sequence of the events whose occurrence has been detected.

26. (Currently amended) Apparatus according to claim 18, wherein the processor is arranged to compute the successive reachable sets by determining a first set among the reachable sets, disjoint from the initial set, such that all of the states in the first set are reached from the ~~initial states~~ at least one initial state in a first cycle of the transition relation, followed by determining the successive reachable sets, following the first set, such that all the states in each of the sets are reached from the states in ~~the~~ a preceding set in a successive cycle of the transition relation, and so that each of the sets is disjoint from the initial set and from ~~the~~ other sets determined before it.

27. (Original) Apparatus according to claim 26, wherein the processor is arranged to produce the partial trace by selecting one of the states from each of at least some of the successive reachable sets.

28. (Previously presented) Apparatus according to claim 27, wherein the processor is arranged to select the states from each of the at least some of the successive sets by choosing, for each of the states, a predecessor state among the states in the preceding set until the state on the trace in the first set is found, and choosing, in the initial set, the predecessor state to the state in the first set.

29. (Currently amended) Apparatus according to claim 18, and wherein the processor is further arranged, upon determining that the intersection exists between the target ~~sets~~ set and one of the reachable sets, to produce a complete trace from the at least one target state through the states in the reachable sets to the at least one initial state.

30. (Original) Apparatus according to claim 29, wherein the processor is arranged to produce the complete trace so that all the events occur along the trace in the specified sequence.

31. (Currently amended) Apparatus according to claim 18, wherein the ~~path~~ specification of the path comprises a property to be fulfilled by the at least one target state.

32. (Previously presented) Apparatus according to claim 31, wherein the property comprises a condition that is expected to be true over all of the reachable states of the hardware system under study, and wherein the condition is false in the at least one target state.

33. (Previously presented) Apparatus according to claim 31, wherein the property comprises a condition representing a desired behavior of the hardware system under study, such that the condition is fulfilled in the at least one target state.

34. (Original) Apparatus according to claim 31, wherein the processor is arranged to test the property while computing the successive reachable sets, and to cease to compute the sets when the intersection is found to exist.

35. (Previously presented) A computer software product, comprising a computer-readable medium in which program instructions are stored, which instructions, when read by a computer, cause the computer to receive a model that defines states of a hardware system under study and a transition relation among the states, and to receive a specification of a path to be found through the states of the hardware system under study from an initial set that comprises at least one initial state among the states of the hardware system to a target set that comprises at least one target state among the states of the hardware system, such that a specified sequence of events is to occur on the path between the at least one initial state and the at least one target state, and which cause the computer to compute, beginning from the initial set, successive reachable sets comprising the states of the hardware system that are reachable from the initial set along the path, such that in the successive reachable sets the events occur in the specified sequence, and to determine whether an intersection exists between one of the reachable sets on the path and the target set, and when the intersection

is not found to exist, to produce a partial trace along the path between the at least one initial state and a termination state in which at least one of the specified events occurs.

36. (Previously presented) A product according to claim 35, wherein the specification of the path comprises a definition of the events in terms of transitions among the states of the hardware system under study.

37. (Previously presented) A product according to claim 36, wherein the events are defined in terms of the transitions such that in the specified sequence of events, at least two consecutive transitions are separated by more than one cycle of the transition relation.

38. (Original) A product according to claim 36, wherein the instructions cause the computer to build a non-deterministic automaton based on the transitions, and to compute the reachable sets using the automaton.

39. (Currently amended) A product according to claim 38, wherein the instructions cause the computer to determine Boolean conditions corresponding respectively to the transitions, and to detect ~~the~~ occurrence of the events ~~comprises~~ by testing the Boolean conditions.

40. (Previously presented) A product according to claim 35, wherein the instructions cause the computer to detect

occurrence of the events in the specified sequence while computing the successive reachable sets.

41. (Original) A product according to claim 40, wherein the instructions cause the computer to inform a user upon detecting occurrence of the events.

42. (Previously presented) A product according to claim 40, wherein the instructions cause the computer to produce the partial trace by choosing the termination state to be one of the states in which a final event occurs in the specified sequence of the events whose occurrence has been detected.

43. (Currently amended) A product according to claim 35, wherein the instructions cause the computer to compute the successive reachable sets by determining a first set among the reachable sets, disjoint from the initial set, such that all of the states in the first set are reached from the ~~initial states~~ at least one initial state in a first cycle of the transition relation, followed by determining the successive reachable sets, following the first set, such that all the states in each of the sets are reached from the states in ~~the~~ a preceding set in a successive cycle of the transition relation, and so that each of the sets is disjoint from the initial set and from ~~the~~ other sets determined before it.

44. (Original) A product according to claim 43, wherein the instructions cause the computer to produce the partial trace by selecting one of the states from each of at least some of the successive reachable sets.

45. (Previously presented) A product according to claim 44, wherein the instructions cause the computer to select the states from each of the at least some of the successive sets by choosing, for each of the states, a predecessor state among the states in the preceding set until the state on the trace in the first set is found, and choosing, in the initial set, the predecessor state to the state in the first set.

46. (Currently amended) A product according to claim 35, and wherein the instructions further cause the computer, upon determining that the intersection exists between the target ~~sets~~ set and one of the reachable sets, to produce a complete trace from the at least one target state through the states in the reachable sets to the at least one initial state.

47. (Original) A product according to claim 46, wherein the instructions cause the computer to produce the complete trace so that all the events occur along the trace in the specified sequence.

48. (Currently amended) A product according to claim 35, wherein the ~~path~~ specification of the path comprises a property to be fulfilled by the at least one target state.

49. (Previously presented) A product according to claim 48, wherein the property comprises a condition that is expected to be true over all of the reachable states of the hardware system under study, and wherein the condition is false in the at least one target state.

50. (Previously presented) A product according to claim 48, wherein the property comprises a condition representing a desired behavior of the hardware system under study, such that the condition is fulfilled in the at least one target state.

51. (Original) A product according to claim 48, wherein the instructions cause the computer to test the property while computing the successive reachable sets, and to cease to compute the sets when the intersection is found to exist.